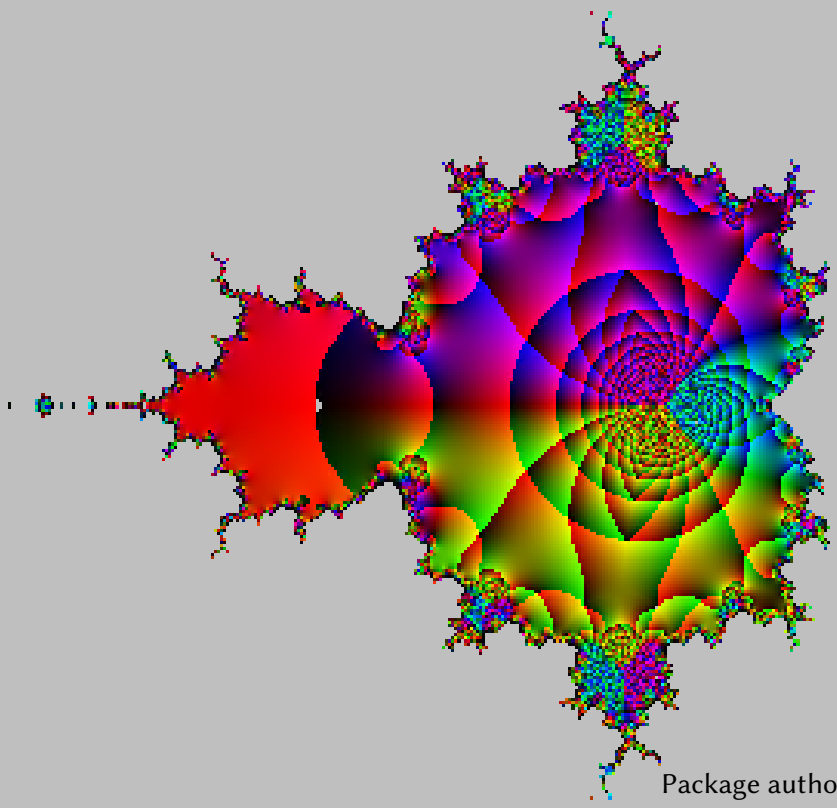
Domain Coloring of complex functions version 0.04

August 29, 2024



Package author(s):
Herbert Voß

*

Contents

1 Introduction 2
1.1 Loading the package . . . 2
1.2 Using the macro . . . 2
1.3 Options . . . 2

2 Examples 3
2.1 The default with function $f(z) = z$ and $f(z) = 1/z$ . . . 3
2.2 Defining domain, color mode, resolution and filename . . . 3
2.3 Option for `\includegraphics` . . . 4
2.4 Higher resolution . . . 4
2.5 hsv to rgb conversion . . . 5
2.6 External function definition . . . 6

References 8

1 Introduction

This package works only with `lualatex` and the option `shell-escape`! It creates an intermediate external EPS-file, which is automatically converted with `epstopdf`. The pdf is in the end imported by the macro `\includegraphics`.

1.1 Loading the package

The package `domaincoloring` creates acolored interpretation of the domain of a complex function. The package itself has no options and should be loaded as usual:

```
\usepackage{domaincoloring}
```

The needs the following Lua modules:

- `domaincoloring.lua` the main module
- `domaincoloring-complex-numbers.lua` for complex math operations
- `domaincoloring-functions.lua` for predefined complex functions

The function module has to be managed by the user himself, if needed.

1.2 Using the macro

There is only one macro which does the external call of the Lua program `domaincoloring.lua`. This program creates the image which is then included into the document. The LaTeX-run needs the `--shell-escape` option to allow the external run of the program to convert the created ppm-file into a png file..

```
\DomainColoring[options]{complex function in Lua notation}
```

Every math function has to be preceeded by `cmath` if it has a complex argument.

1.3 Options

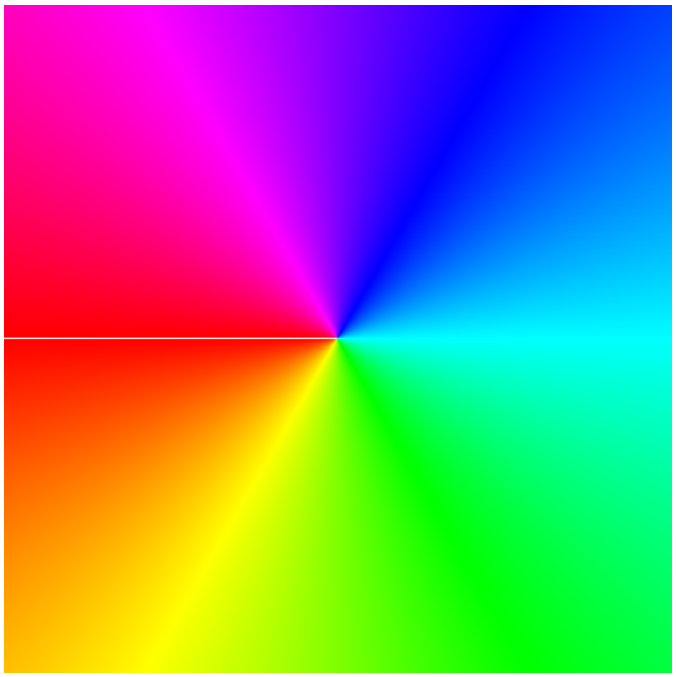
name	*value*	*meaning*
`domain`	-2,2,-2,2	the (re,im)-coordinates for the complex system

`resolution`	500	the number of steps for the re,im interval. One value will be for both axes. Two values like `{500,600}` for real axis and imaginary axis,
`filename`	`\jobname-tmp`	name of the created external file, must be unique
`funcName`	{}	corresponding to external file
`grfOptions`	scale=0.5	optional arguments for `\includegraphics`
`hsvrgb`	phi+pi,1,r	for the conversion into rgb
`bgcolor`	-1	change color to white as background for all values $-1 \le r+g+b \le$ bgcolor
`invers`	false	inverted colors with $color = \lvert color - 255 \rvert$
`force`	true	With `force=false` an existing pdf file will be used without calculating a new one.
`grid`	false	draw a grid with one dashed subgrid at 0.5

2 Examples

2.1 The default with function $f(z) = z$ and $f(z) = 1/z$

```
\DomainColoring{z}   % default filename \jobname-tmp.png
```



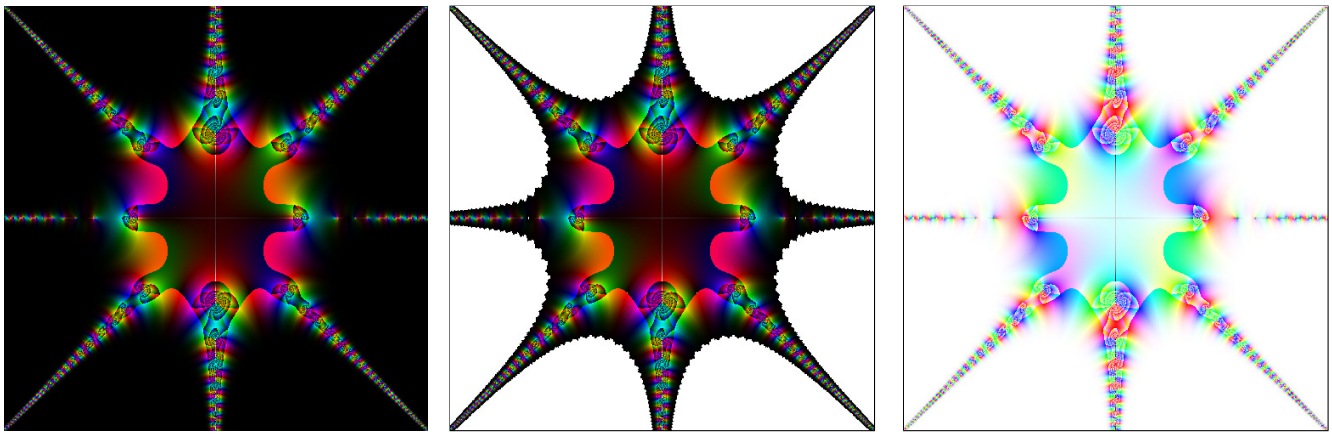
2.2 Defining domain, color mode, resolution and filename

$$f(z) = \cos(z)/\sin(z^4 - 1) \tag{1}$$

in Lua-notation: `cmath.cos(z)/cmath.sin(z^4-1)`. All complex functions must be preceeded by `cmath.`. For real functions the prefix `math.` can be omitted.

```
\DomainColoring[domain={-2.5,2.5,-2.5,2.5},resolution=500,hsvrgb={phi,1,r},
grfOptions={width=0.32\linewidth},
    filename=\jobname-tmp1a]{cmath.cos(z)/cmath.sin(z^4-1)}
\hfill
\frame{\DomainColoring[domain={-2.5,2.5,-2.5,2.5},resolution=500,
  bgcolor=3,hsvrgb={phi,1,r},grfOptions={width=0.32\linewidth},
    filename=\jobname-tmp1b]{cmath.cos(z)/cmath.sin(z^4-1)}}
\hfill
```

```
\frame{\DomainColoring[domain={-2.5,2.5,-2.5,2.5},resolution=500,invers=true,
   hsvrgb={phi,1,r}, grfOptions={width=0.32\linewidth},
    filename=\jobname-tmp1c]{cmath.cos(z)/cmath.sin(z^4-1)}}
```

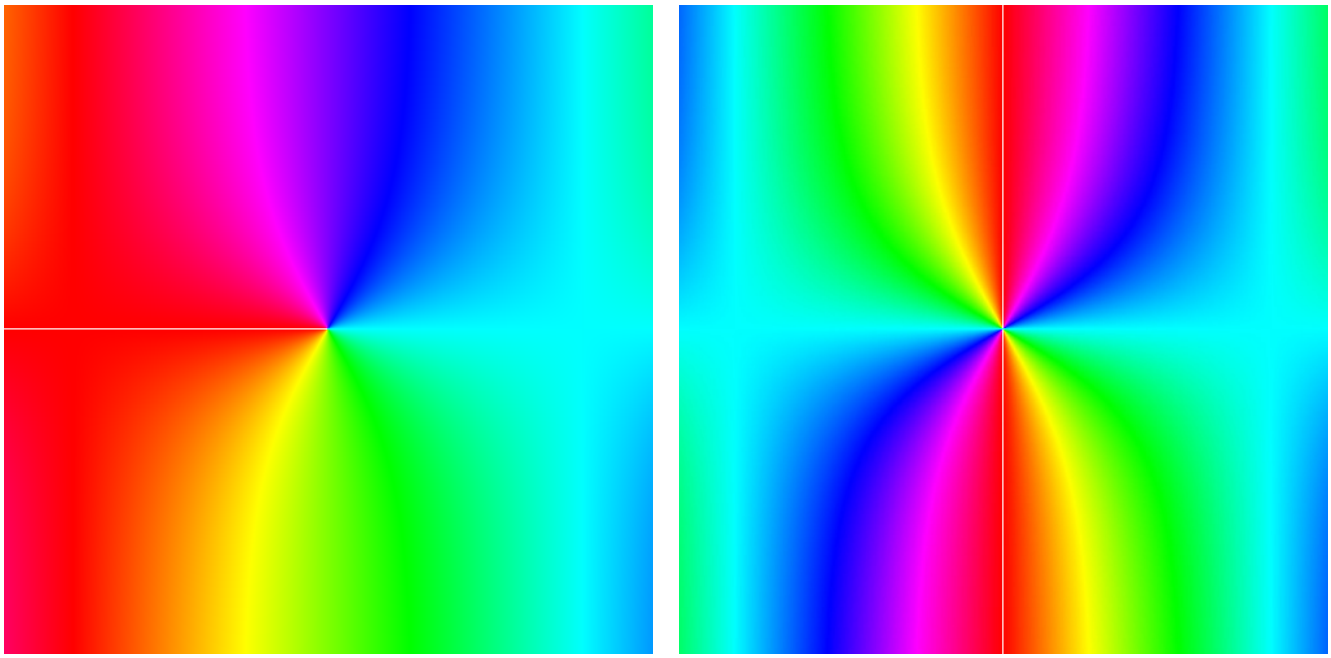


With `bgcolor=<int>` all RGB-combinations with a sum $R + G + B <=$ int are set to the color white.

2.3 Option for `\includegraphics`

With `grfOptions` one can define optional arguments for `\includegraphics`:

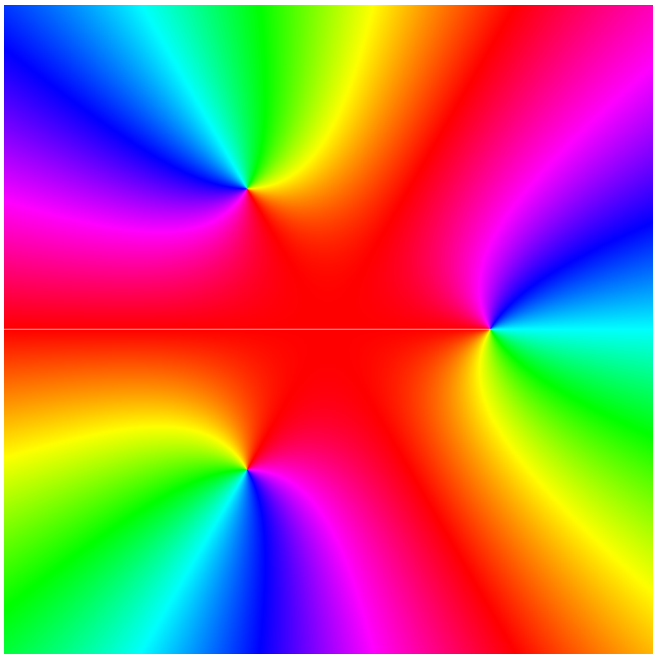
```
\DomainColoring[grfOptions={width=0.49\linewidth},filename=\jobname-tmp2a]{cmath.sin(z)}\hfill
\hfill
\DomainColoring[grfOptions={width=0.49\linewidth},filename=\jobname-tmp2b]{cmath.sin(0.9*z)*cmath.sin(z)}
```

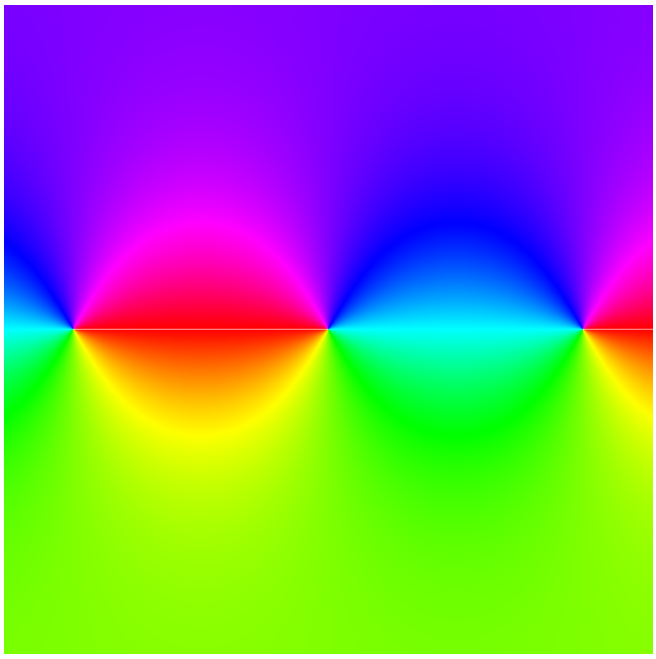


2.4 Higher resolution

The resolution is more or less the number of pixels for the given domain. It is possible to have different values for the two coordinates. Is only one value for `resolution` given, then it is for both axes.

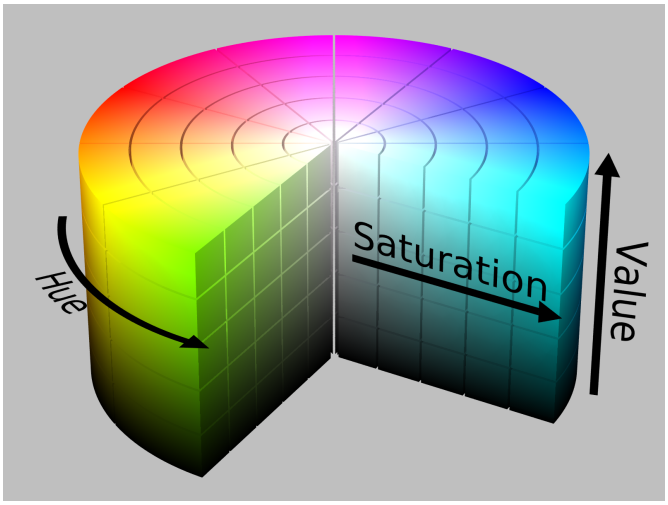
```
\DomainColoring[filename=\jobname-tmp3,resolution=1000,grfOptions={width=0.49\linewidth}]{z^3-1}
\hfill
\DomainColoring[filename=\jobname-tmp4,resolution=1000,
  grfOptions={width=0.49\linewidth}]{(z+1)^2*(z-1)/((z+i)*(z-i)^2)}
```





2.5 hsv to rgb conversion

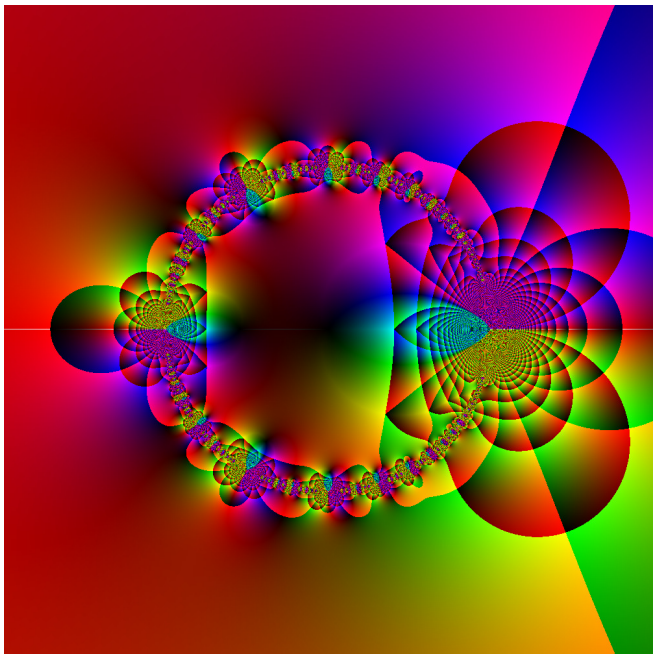
The color model (Wikipedia):

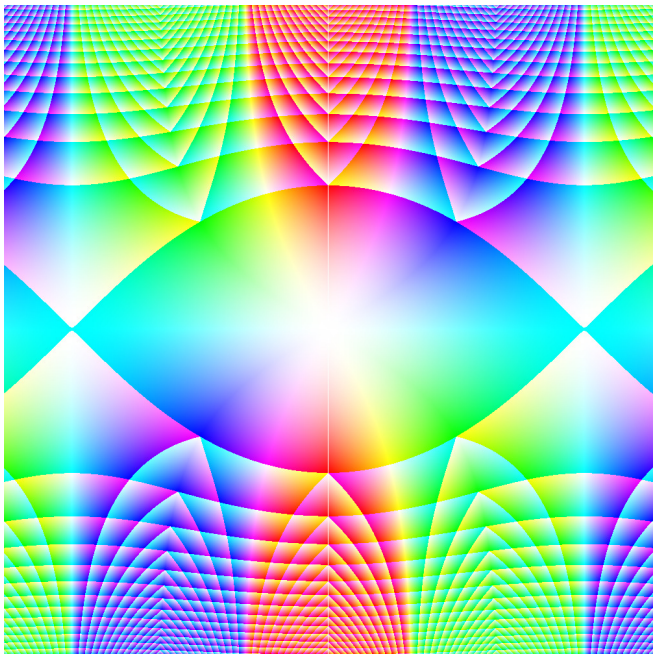


http://en.wikipedia.org/wiki/File:HSV_color_solid_cylinder_alpha_lowgamma.png

The complex number $z = x+iy$ is converted into its trogonometrical representation $x = r\cdot\cos\phi$ and $y = r\cdot\sin\phi$ with $r = \sqrt{x^2+y^2}$. The values r and ϕ are now taken as values for the hsv color model with a constant second value for saturation. ϕ is used for hue. For example: `hsvrgb=phi,1,r`, which gives

```
\DomainColoring[filename=\jobname-tmp5a,resolution=1000,
   grfOptions={width=0.49\linewidth},hsvrgb={phi,1,r}]{cmath.sin(z)*cmath.sin(0.99*z)}
\hfill
\DomainColoring[filename=\jobname-tmp5b,resolution=1000,,hsvrgb={phi,r,1},
   grfOptions={width=0.49\linewidth}]{cmath.sin(z)*cmath.sin(0.99*z)} % the default
```



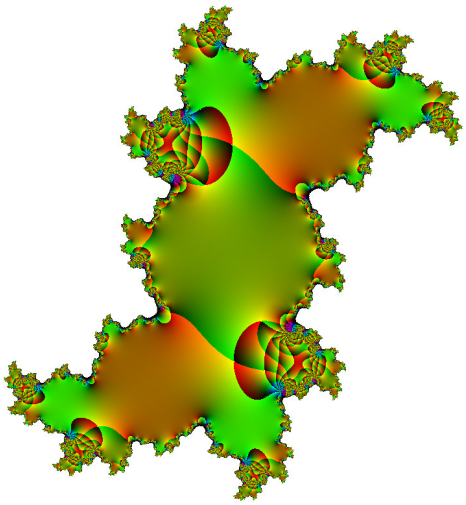


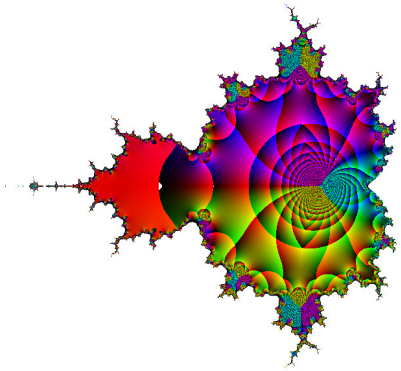
The optional argument `hsvrgb` must define three values which can use the arguments phi and r in any mathematical combination. It must only be compatible to the Lua math conventions, e.g. `hsvrgb={phi+2,0.5,2/r}`

2.6 External function definition

The already existing file `domaincoloring-functions.lua` collects sone definitions of complex functions $f(z)$, which can be used from inside LaTeX with the optional argument `funcName<Lua function name>`. In this case the mandatory argument of `\DomainColoring` has no meaning and can be empty.

```
\DomainColoring[domain={-1.5,1.5,-1.5,1.5},resolution={1001,1001},hsvrgb={phi,1,1/r},
  filename=\jobname-tmp4.png, grfOptions={width=0.49\linewidth},funcName=f12,bgcolor=5]{}
\hfill
\DomainColoring[domain={-2.5,1.5,-2,2},resolution={1001,1001},hsvrgb={phi,1,1/r},
  filename=\jobname-tmp5.png, grfOptions={width=0.49\linewidth},funcName=f13,bgcolor=5]{}
```





The contents of the function file of the current version of `domaincoloring` is::

```
-- $Id: domaincoloring-functions.lua 968 2024-08-29 11:44:46Z herbert $

kpse.set_program_name("luatex")

function f0(z)
   return cmath.sin(1/z)*cmath.cos(1/z)/z^3+1/z
end

function f1(z)
    return cmath.cos(z)/cmath.sin(z^4-1)
end

function f2(z)
  local c = complex(1,-1)
  local d = complex(0,0.28)
  return cmath.cos(c^2*z^2)/cmath.cos(c*(z-d))
end

function f3(z)
  return z*(z+i)^2/(z-i)^2
end

function f4(z)
  if abs(z) < 0.1 then
    return complex(0.001,0.001)
  else
    return cmath.sin(1/(z*z))
  end
end

function f5(z)
  return cmath.sqrt(1-1/(z*z)+z^3)
end

function f9(z)
  local c = complex(1,-1)
  local d = complex(1,1)
  return z^2*c^2*(z*c-1-i)/(z*c-2*d)
end

function f10(z)
  local sum = complex(0,0)
  for n=1,20 do
      sum = sum + z^n/(1-z^n)
  end
  return sum
end

function f11(z)
  local iterateNo = 3
  for n=1,iterateNo do
    z = z^2
  end
  return z
end
```

```
function f12(z)   -- julia
  local iterateNo = 15
  for n=1,iterateNo do
    z = z^2 + complex(0.25,-0.5)
  end
  return z
end

function f13(z)   -- mandelbrot
  local iterateNo = 15
  local c = z
  z = complex(0,0)
  for n=1,iterateNo do
    z = z^2 + c
  end
  return z
end

function f14(z)
  local iterateNo = 5
--  local c = z
--  z = complex(0,0)
  for n=1,iterateNo do
    z = cmath.sin(z)*cmath.sin(0.8*z) -- + c
  end
  return z
end

function f15(z)
  local alpha = 4
  local C0 = complex(1,0)
  local C1 = 2 * alpha * z
  for n = 2,20 do
    C = (2*z*(n+alpha-1)*C1 - (n+2*alpha-2)*C0)/n
    C0 = C1
    C1 = C
  end
  return C
end
```

References

[1] Juan Carlos Ponce Campuzano. *Dynamic Mathematics. Domain Coloring – Visualizing Complex Functions.* July 15, 2018. URL: https://www.dynamicmath.xyz/domain-coloring/ (visited on 08/23/2024).

[2] Konstantin Poelke and Konrad Polthier. *Domain Coloring of Complex Functions.* Aug. 18, 2024. URL: https://www.mi.fu-berlin.de/en/math/groups/ag-geom/publications/db/ieee_article_old_low_v3_1.pdf (visited on 08/18/2024).

[3] vismath. *Thema Domain Coloring.* Aug. 18, 2024. URL: https://www.vismath.eu/de/blog/domain-coloring/ (visited on 08/18/2024).

[4] WIKIPEDIA. *DOmain Coloring.* Aug. 18, 2024. URL: https://en.wikipedia.org/wiki/Domain_coloring (visited on 08/18/2024).

Index

B

bgcolor, 3f

D

domain, 2
domaincoloring-functions.lua, 6
\DomainColoring, 6
domaincoloring, 6

E

epstopdf, 2

F

File
- domaincoloring-functions.lua, 6
- epstopdf, 2

filename, 3
force, 3
funcName, 3, 6

G

grfOptions, 3f
grid, 3

H

hsvrgb, 3, 6

I

\includegraphics, 2ff
invers, 3

J

\jobname-tmp, 3

K

Keyword
- bgcolor, 3f
- domain, 2
- filename, 3
- force, 3
- funcName, 3, 6
- grfOptions, 3f
- grid, 3
- hsvrgb, 3, 6
- invers, 3
- resolution, 3f

M

Macro
- \DomainColoring, 6
- \includegraphics, 2ff
- \jobname-tmp, 3

P

Package option
- shell-escape, 2

Package
- domaincoloring, 6

R

resolution, 3f

S

shell-escape, 2